AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method for amortizing a critical path computations in a circuit comprising:

   unrolling a data flow graph representing said circuit into a plurality of clock cycles; and

   simulating said circuit in said plurality of clock cycles on a computer,

   wherein simulating further comprises reducing a difference between said critical path and a shortest path in said data flow graph.

2. (Canceled)

3. (Currently Amended) The method of claim 1 [[2]], wherein said step of reducing further comprises:

   compacting one or more computations from said plurality of clock cycles in a processor.

4. (Currently Amended) The method of claim 1 [[2]], wherein said step of unrolling further comprises:

   eliminating one or more flip-flops between one or more boundaries within said plurality of clock cycles.

5. (Currently Amended) The method of claim 1 [[2]], wherein said step of unrolling further comprises:

   eliminating one or more latches between one or more boundaries within said plurality of clock cycles.

6. (Original) The method of claim 1, wherein said computer has a plurality of processors.

7. (Original) The method of claim 1, wherein said computer has a plurality of simulation processors, wherein said simulation processors include a communication network interconnecting said simulation processors for data communication, said simulation

processors further including a synchronization network interconnecting said simulation processors for synchronizing execution there between.

8. (Original) The method of claim 1, wherein said step of simulating further comprises:
delaying evaluation of one or more logic elements within said plurality of clock cycles, thereby creating a timing slack for inter-processor communication.

9. (Original) The method of claim 3, wherein said step of reducing further comprises:
using a first processor wherein said processor computes said critical path and a non-critical path in a said plurality of clock cycles.

10. (Original) The method of claim 1, further comprising:
compacting said plurality of clock cycles into a single clock cycle.

11. (Currently Amended) A critical path computation amortizer for a circuit comprising:
a data flow graph unroller configured to represent said circuit into a plurality of clock cycles; and
a simulator configured to simulate said circuit in said plurality of clock cycles on a computer,
wherein the simulator further comprises a reducer configured to reduce a difference between said critical path and a shortest path.

12. (Canceled)

13. (Currently Amended) The critical path computation amortizer of claim 11 [[12]], wherein said reducer further comprises:
a compactor configured to compact one or more computations from said plurality of clock cycles in a processor.

14. (Currently Amended) The critical path computation amortizer of claim 11 [[12]], wherein said unroller further comprises:
an eliminator configured to eliminate one or more flip-flops at one or more boundaries within said plurality of clock cycles.

15. (Currently Amended) The critical path computation amortizer of claim 11 [[12]], wherein said unroller further comprises:

    an eliminator configured to eliminate one or more latches at one or more boundaries within said plurality of clock cycles.

16. (Currently Amended) The critical path computation amortizer 11 [[12]], wherein said computer has a plurality of processors.

17. (Original) The critical path computation amortizer of claim 11, wherein said computer has a plurality of simulation processors, wherein said simulation processors include a communication network interconnecting said simulation processors for data communication, said simulation processors further including a synchronization network interconnecting said simulation processors for synchronizing execution therebetween.

18. (Original) The critical path computation amortizer of claim 11, wherein said simulator is further configured to delay evaluation of one or more logic elements in said plurality of clock cycles, thereby creating a timing slack for inter-processor communication.

19. (Original) The critical path computation amortizer of claim 13, wherein said reducer further comprises:

    a feed-back configured to use a first processor wherein, said first processor computes said critical path and a non-critical path in said plurality of clock cycles.

20. (Original) The critical path computation amortizer of claim 11, further comprising:

    a scheduling compactor configured to compact said plurality of clock cycles into a single clock cycle.

21. (Currently Amended) A computer program product comprising:

    a computer usable medium having computer readable program code embodied therein configured to amortize a critical path computation in a circuit, said computer program product comprising:

    computer readable code configured to cause a computer to unroll a data flow graph representing said circuit into a plurality of clock cycles;

computer readable code configured to cause a computer to simulate said circuit in said plurality of clock cycles on a computer, by at least causing the computer to reduce a difference between said critical path and a shortest path.

22. (Canceled)

23. (Currently Amended) The computer program product of claim 21 [[22]], wherein said computer readable code configured to cause a computer to reduce further comprises:
computer readable code configured to cause a computer to compact one or more computations from said plurality of clock cycles in a processor.

24. (Currently Amended) The computer program product of claim 21 [[12]], wherein said computer readable code configured to cause a computer to unroll further comprises:
computer readable code configured to cause a computer to eliminate one or more flip-flops at one or more boundaries within said plurality of clock cycles.

25. (Currently Amended) The computer program product of claim 21 [[12]], wherein said computer readable code configured to cause a computer to unroll further comprises:
computer readable code configured to cause a computer to eliminate one or more latches at one or more boundaries within said plurality of clock cycles.

26. (Original) The computer program product of claim 21, wherein said computer has a plurality of processors.

27. (Original) The computer program product of claim 21, wherein said computer has a plurality of simulation processors, wherein said simulation processors include a communication network interconnecting said simulation processors for data communication, said simulation processors further including a synchronization network interconnecting said simulation processors for synchronizing execution therebetween.

28. (Original) The computer program product of claim 21, wherein said computer readable code configured to cause a computer to simulate further comprises:

computer readable code configured to cause a computer to delay evaluation of one or more logic elements in said plurality of clock cycles, thereby creating a timing slack for inter-processor communication.

29. (Currently Amended) The computer program product of claim 21 [[20]], wherein said computer readable code configured to cause a computer to reduce further comprises:

computer readable code configured to cause a computer to use a first processor wherein said first processor computes said critical path and a non-critical path in said plurality of clock cycles.

30. (Original) The computer program product of claim 21, further comprising:

computer readable code configured to cause a computer to compact said plurality of clock cycles into a single clock cycle.